OMNIBIUS POCKET COMPUTER REFERENCE

DRM System

The Omnibius Pocket Computer uses Digital Rights Management to protect the intellectual property in the ROM data. The state of the art encryption uses a secret 16-bit key to scramble the ROM data. The ROM must be unscrambled before loading it into memory. To protect the copyright holders, every unscrambled ROM also contains an ASCII copyright notice to deter illegal reproduction of unscrabled ROM data. See the **OMNIBIUS CONTENT PROTECTION REFERENCE** document for implementation details.

CPU Info

The Omnibius CPU has an instruction set that consists of 8-bit and 16-bit instructions. The instruction operands are a part of the instruction. Memory (**MEM**) is addressed in 8-bit (1-byte) segments, but REG_OR_ADDR operand instructions referring to addresses will read/write 16 bits (2 bytes) of memory at once.

The Omnibius Pocket Computer has 16kB of RAM. The ROM cartridges can come in sizes of 2kB, 4kB, 8kB, and 16kB.

Register name	Size (bits)	Reference (binary)
IP	16	N/A
REG_A	16	00
REG_B	16	01

Instructions with **REG_OR_ADDR** operands can target their corresponding addresses by adding 0b10 to the reference. For example, a XOR instruction written as **00 11 00 01** would XOR REG_B with REG_A and store the value in REG_A, but the instruction written as **00 11 10 01** would XOR REG_B with MEM[REG_A] and store the value in MEM[REG_A].

The Instruction Pointer (IP) begins at 0 and refers to an offset in MEM. Before execution, the contents of the ROM shall be copied to MEM at address 0. The IP will be incremented after executing every instruction by the length of the instruction.

CPU Instructions

Mnemonic	Summary
XOR	Exclusive OR
XORV	Exclusive OR Value
SCREEN	Output Screen Graphics
NOTE	Emit MIDI Note
HLT	Halt

XOR - Exclusive OR

Opcode (binary)	Operand 1 (A)	Operand 2 (B)
00 11 AA BB	Target REG_OR_ADDR	Source REG_OR_ADDR

Description

XORs the Source and Target, and stores the result in Target.

XOR - Exclusive OR Value

Opcode (binary)	Operand 1 (A)	Operand 2 (B)

1A BBBBBBBBBBBBBB	Target REG	Source LITERAL (14 bits)

Description

XORs the literal 14-bit Source value with the Target register, and stores the result in the Target register. This instruction can only target a register, not a memory address.

SCREEN - Output Screen Graphics

Opcode (binary)	Operand 1 (A)	Operand 2 (B)
00 01 AA BB	Address REG_OR_ADDR	Mode LITERAL (2 bits)

Description

Draws the graphics at the memory address Address to the screen using an algorithm chosen by Mode. The graphics output is always 1-bit.

Mode (binary)	Name	Dimensions	Description
00	Bitplane	64x64	In this mode, the pixels will be copied to the output linearly from Address, horizontally-first. This mode outputs 512 bytes of memory.
01	Tiled	64x64	In this mode, 16 bits will be read at a time and drawn as 4x4 tiles on the screen, horizontally-first. The next tile drawn will be 4 pixels over. The next line drawn will be 4 pixels down. This mode outputs 512 bytes of memory.
10	Bitplane	128x128	In this mode, the pixels will be copied to the output linearly from Address, horizontally-first. This mode outputs 2048 bytes of memory.
11	Tiled	128x128	In this mode, 16 bits will be read at a time and drawn as 4x4 tiles on the screen, horizontally-first. The next tile drawn will be 4 pixels over. The next line drawn will be 4 pixels down. This mode outputs 2048 bytes of memory.

NOTE - Emit MIDI Note

Opcode (binary)	Operand 1 (A)	Operand 2 (B)
00 10 AA BB	Note REG_OR_ADDR	Duration REG_OR_ADDR

Description

Plays the MIDI note Note for Duration milliseconds.

HLT - Halt

Opcode (binary)	Operand 1 (A)	Operand 2 (B)
00 00 00 00	N/A	N/A

Description

Halts the execution of the program. No instructions should be executed beyond this point.